Secure Billing for Mobile Information Services in UMTS

K M Martin[1], B Preneel[1], C J Mitchell[2], H J Hitz[3], G Horn[3], A Poliakova[3], P Howard[4]

([1]) = Katholieke Universiteit Leuven, ESAT-COSIC, B-3001 Heverlee, Belgium
{keith.martin, bart.preneel}@esat.kuleuven.ac.be
([2]) = Royal Holloway, University of London, ISG, Egham, TW20 0EX, UK.
c.mitchell@rhbnc.ac.uk
([3]) = Siemens AG, Corporate Technology, D-81730-München, Germany.
{hans-joachim.hitz, guenther.horn, alla.poliakova}@mchp.siemens.de
([4]) = Vodafone Ltd, CSAD, 2-4 London Road, Newbury, RG14 1JX, UK
peter.howard@vf.vodafone.co.uk

Abstract. This paper presents solutions developed in the ACTS ASPeCT project for advanced security features in UMTS. In particular, a secure billing scheme for value-added information services using micropayments is presented. The solutions will be validated in a trial to be conducted over an experimental UMTS platform.

1 Introduction

It is clear that adequate security features must form an integral part of a mobile telecommunications system. In second generation systems such as GSM and DECT, security features based on cryptographic techniques have been included in a systematic way for the first time [1, 2]. Their success is undeniable: second generation systems are much less susceptible to fraud than their predecessors. However, the increasing, and increasingly diverse, demand for security by users, operators and regulatory bodies calls for more advanced security features in third generation systems, such as the *Universal Mobile Telecommunications System* (UMTS). It is the goal of the ACTS *Advanced Security for Personal Communications Technologies* (ASPeCT) project to specify such advanced features, propose solutions, and test these solutions in demonstrations and trials.

Some of the advanced security features to be provided in UMTS will be made possible by the use of more powerful smart card technology not yet available for second generation systems. This technology, together with the use of suitably adapted security mechanisms, will make the use of so-called public key techniques in mobile systems practical for the first time.

The infrastructure to support public key techniques will be provided by *Trusted Third Parties* (TTPs). The services provided by TTPs will include the certification

and management of public keys. A TTP infrastructure will enable the provision of non-repudiation services based on digital signatures, opening up the possibility for secure billing services over UMTS. In this paper we focus on a scheme to bill the mobile user for value-added information services. Such services are expected to become increasingly important as current networks evolve towards UMTS.

In the remainder of this paper we describe some background to the work conducted by ASPeCT on secure billing, and provide protocol descriptions for a scheme to be trialled over an experimental UMTS platform provided by the ACTS *Experiments on the Deployment of UMTS* (EXODUS) project.

2 Mobile Information Services

A future mobile user will be offered a much greater variety of services than those available in today's networks. However, there will still be a distinction between basic tele- and bearer- services, such as traditional telephony, video telephony or high speed data services, and services offering added value to the user, such as the provision of information. In ASPeCT we concentrate on schemes to bill users for such *Value-Added Services* (VASs).

It is expected that the number and variety of *Value-Added Service Providers* (VASPs) will greatly increase while current networks are evolving towards UMTS. One reason for this is that users will increasingly possess terminals with much greater processing and display capabilities than those of today's mainly speech-orientated terminals. Personal mobile communicators will integrate the functions of a mobile phone, and of a laptop or palmtop personal computer. These devices may be used to access information of a much more complex nature than that available to users of VASs in mobile systems today. Instead of being restricted to the character-oriented display of his terminal, the user will have more advanced capabilities, such as the ability to display hypertext documents with graphics. So, instead of textual information on the nearest hotels, he may be able to browse a street map of his surroundings giving the location and images of the nearby accommodation. It is clear that an appropriate charging scheme must be made available in order to allow users to securely pay VASPs for these information services.

A crucial element in our model is the *User Identity Module* (UIM), which is contained in a smart card held by the user and issued by the user’s UMTS service provider. This smart card will be multi-functional, and will contain the security procedures to access basic UMTS services as well as advanced features to pay for value-added services.

3 Certification Infrastructure

The role of TTPs in supporting security services by means of a wide range of cryptographic techniques is recognised in a large variety of application domains.

Within the ASPeCT project we are primarily concerned with the use of TTPs to support mobile telecommunications security services which use public key cryptography.

In support of secure billing for value-added information services, a TTP infrastructure will facilitate electronic transactions between a large number of mobile users and VASPs in a UMTS environment, where each pair of users and VASPs who wish to conduct transactions do not have to have any previous security relationship. To support the secure billing services, mobile users and VASPs will each register with TTPs acting as *Certification Authorities* (CAs), which will certify and manage public keys for them.

A non-standard, compact *public key certificate* format has been chosen in ASPeCT because both storage space on a smart card and bandwidth on the air interface are strictly limited. Each certificate consists of two parts: a *certificate type identifier* and a signed *certificate information sequence*. We have two types of certificates in ASPeCT, depending on the signature mechanism used. The first type uses RSA-signatures based on ISO/IEC 9796-2 [3], where the signed certificate information sequence includes both a signed recoverable message string and a non-recoverable part. The second type uses AMV-signatures based on ISO/IEC 14888-3 [4], where the signed certificate information sequence is the message string itself concatenated with a signature of the message string. Both types of certificates are used in the ASPeCT secure billing trial.

The certificate information sequence profile for the ASPeCT secure billing trial ignores most of the options in the general ASPeCT certificate information sequence format. The certificate information sequence includes basic certificate information and a set of extended attributes providing other optional information about both the subject and the issuer.

4 Requirements On Charging Schemes

Charging for today's VASs consists of a charge for the basic service and a premium for the added value. Both charges are based on the duration of the call. In the future, due to the greater variety of services on offer, more *flexible* charging schemes for the premium would be desirable. Flexibility relates to the parameters which determine the charge (in addition to the duration of the call, the charge could depend on the amount of data transferred), to the variety of different possible tariffs, and to the ease with which a certain tariff can be changed.

The value of a particular piece of information retrieved by a user from a VASP at any one time may be quite small. Charging schemes should thus not require a large financial overhead in order to process the charge. Furthermore, the scheme should have a performance compatible with the requirements of a mobile system. In short, the charging scheme must be *efficient.*

It is expected that the evolution of current mobile systems towards UMTS will also see the emergence of many new network operators, UMTS service providers, and VASPs, which may have serious implications for the trust relationships between

them. Thus, the charging scheme must be *secure* against cheating (such as overcharging by the VASP or underpaying by the user), and the parties involved should have the assurance that justified claims relating to charges can be proved and that unjustified claims cannot be successfully made. This is often called *incontestable charging*.

ASPeCT has demonstrated a charging scheme for VASs in UMTS which satisfies the above requirements. The charging scheme is implemented as a credit-based payment scheme using *micropayments*, although it is possible to use the scheme on a pre-paid basis.

5 Micropayments

Micropayment schemes are electronic payment schemes that are proposed explicitly for the payment of items costing very small amounts. The interest in such schemes is largely based on the need to develop efficient methods of electronically paying for items such as information on the World-Wide Web.

In most micropayment schemes a *buyer* purchases goods from a *vendor*, and the transaction may or may not also require the participation of a *broker* who may be in contact with a *bank*, or indeed in some cases be a bank. In our payment model, the mobile user is the buyer, the VASP is the vendor and the UMTS service provider, and a TTP acting on its behalf, is the broker.

There are a couple of general design assumptions that characterise micropayment schemes.

1. The cost of communication and processing costs of a micropayment should be kept as low as possible, since otherwise it may not be economical to collect the charge for a micropayment at all.
2. Since potential losses over short periods in a micropayment scheme are low, it may be possible to sacrifice the full security requirements of a payment protocol in order to gain increases in efficiency of operation and decreases in cost. Thus large scale abuse of a micropayment scheme should be prevented, but limited small scale fraud may be worth tolerating.

A wide variety of micropayment schemes are emerging. In ASPeCT we are implementing a scheme based on Pedersen's Ticks [5]. The ticks concept uses chains of pre-images of a one-way function F to make micropayments. This idea has been independently claimed by several authors. Similar and related schemes include PayWord [6], NetCard [7], Micropayments based on iKP [8] and PayTree [9]. The basic idea in all of these schemes is very simple. The user begins with a secret starting value α, commits to the n-th image $F^n(\alpha)$ of the starting value by digitally signing it, and then pays for the i-th micropayment (or tick) by releasing the pre-image $F^{n-i}(\alpha)$. This is essentially the concept first developed by Lamport [10] for one-time passwords, and by Winternitz (see Merkle [11]) for one-time signatures. The major advantage of this technique is that computationally expensive public key signatures are avoided during the payment mechanism, and replaced by computations of a one-way function.

6 Charging Model

The relationship between the entities in our pre-paid charging scheme is illustrated in Fig. 1. The VASP provides information to the user and sends corresponding *charge requests*. As the information is being provided, the user pays for the service by sending *micropayment tokens*. The VASP is able to check the validity of these tokens based on a certificate of the user's credentials issued by the UMTS service provider, or an appropriate TTP acting on behalf of the UMTS service provider. As part of the off-line payment clearance procedure, the VASP forwards *billing information* proving the claims on the user to the user's UMTS service provider, who in turn bills the user as part of the regular subscriber billing process. In return the user sends a *payment* to the UMTS service provider, who forwards the due share to the VASP.

Note that the communication between the user and the VASP is the only on-line communication required in our model.

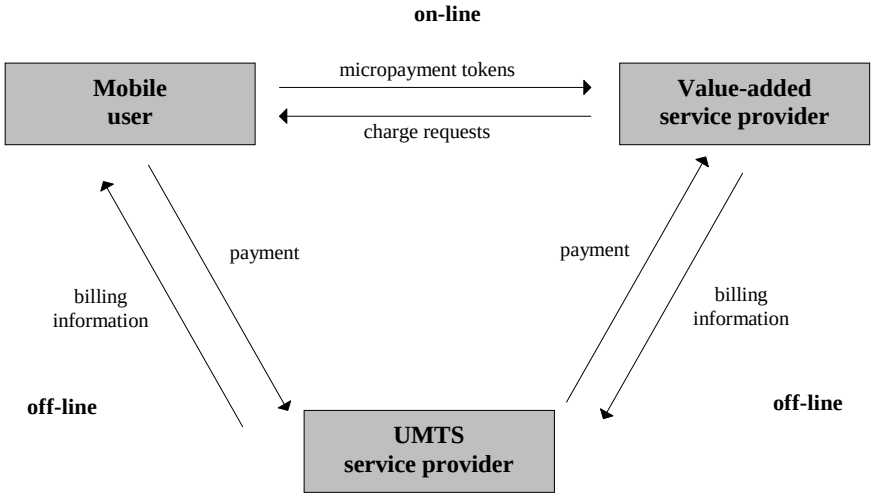

Fig. 1. Charging Model

No previous contact between the VASP and the mobile user's UMTS service provider is required as long as the VASP can satisfactorily verify the user's certificate. However, an existing business arrangement between the VASP and the UMTS service provider would be advantageous in facilitating clearance of the payment. Payment clearance would typically take place at regular intervals, perhaps daily or weekly.

The UMTS service provider plays the role of a *broker*. He provides the user with a means to pay electronically and vouches for the credit-worthiness of the user by issuing a certificate for him. New certificates could be issued periodically, perhaps monthly. If a bill is not paid, then the current certificate can be revoked.

A major advantage of implementing the micropayment scheme in a telecommunications environment is that the banking infrastructure for billing the user and paying the VASP is already in place.

7 Charging Protocol

The charging protocol is split into two phases. The first is the initialisation phase, where the user and the VASP authenticate one another and agree on a session key. At this point the user commits himself to a pre-image chain and a certain tariff for the micropayment scheme by performing a digital signature on corresponding data. In this protocol the VASP may interact with a TTP in order to obtain certificates on the necessary public keys required in the protocol. The authentication scheme used in this protocol is identical to one which has been proposed for UMTS user-to-network authentication [12]. Thus the cost of introducing the micropayment scheme is minimised since existing functions can be reused.

The second phase is the data transfer phase, where the user pays for ticks (which represent unit charges) by releasing pre-images from the chain. The value of one unit charge is agreed upon in the initialisation phase. The ticks serve as proof to the VASP that the user incurred certain charges because only the user could have generated them. These ticks are presented by the VASP to the user's UMTS service provider to clear the charges. The particular efficiency of the scheme stems from the fact that the user may commit himself to a large number of unit charge payments with only one signature. Images of one-way functions are much less expensive to compute and to transmit than signatures.

The two protocols corresponding to the two phases are now presented. For the sake of brevity, we omit a third protocol, the so-called *re-initialisation protocol*, which is used when the user runs out of ticks while the call session is still in progress. In order to ensure the security of the protocol it is important that if any checks involved in processing messages should actually fail, then the protocol, and the associated chargeable service, should be abandoned.

A preliminary version of the charging protocol was described in [13].

Table 1. Algorithms Used In The Protocol

Algorithm	Description
F	A one way function used to calculate $\alpha_T = \mathbf{F^T}(\alpha_0)$ where $\alpha_i = \mathbf{F}(\alpha_{i-1})$ *for* $i=1,...,T$. *F* is selected from a family of one-way functions *f_IV(x)=h4(IV,x)* where *h4* is a one way function and the input to *h4* is an initialisation vector *IV* concatenated with *x*
f	A function mapping points in an elliptic curve cryptosystem onto numbers in the range [0..q-1], where q is the size of the elliptic curve
h1	A one-way hash function used to calculate the session key $K_s = \mathbf{h1}(f((g^{RND_u})^v)) \| RND_v) = \mathbf{h1}(f((g^v)^{RND_u}) \| RND_v)$
h2	A one-way function used to calculate $\mathbf{h2}\,(K \| RND_v \| id_v)$
h3	A collision resistant hash function used to calculate a hash value before signature

Algorithm	Description
	computation
h4	A one way function used in *F*
$\boldsymbol{Sig}_U$	A secret signature transformation owned by the user
$\boldsymbol{Ver}_U$	A verification algorithm corresponding to the signature transformation owned by the user. This algorithm needs the public key (*PK_U* in this case) as input
$\boldsymbol{Sig}_T$	A secret signature transformation owned by the TTP
$\boldsymbol{Ver}_T$	A verification algorithm corresponding to the signature transformation owned by the TTP. This algorithm needs the public key (*PK_T* in this case) as input
$\boldsymbol{\{...\}}_K$	A symmetric encryption algorithm. $\boldsymbol{\{data\}}_K$ means that *data* is encrypted with key *K*

7.1 Authentication And Initialisation Of Charge Ticks Protocol

There are three versions of the authentication and initialisation of charge ticks protocol. The conditions at the start of the protocol determine which of the three versions should be used. The version described here allows mutual authentication between a user and a VASP, in the case where the user and VASP do not share each others' certificates. This version allows transactions to take place between a mobile user and a VASP who have had no previous security relationship.

The goals of this version of the protocol include:

- mutual explicit entity authentication of user and VASP;
- agreement between the user and the VASP on a session key *K* with mutual implicit key authentication;
- mutual key confirmation between the user and the VASP;
- mutual assurance of key freshness;
- non-repudiation by the user of data sent by the user to the VASP;
- confidentiality of the user identity at the interface between the user and the VASP.

In this protocol we assume that the VASP, but not the user, interacts with a TTP. In the general case this could be any TTP belonging to an appropriate certification infrastructure. However, in the following protocol we assume that the VASP communicates with a TTP which acts as the user's CA. Thus, we assume that the TTP in the protocol has a copy of the user's certificate, that the user is able to verify signatures generated by the TTP using an authentic copy of the TTP's public signature verification key, and that the user has an authentic copy of the TTP's public key agreement key. In other scenarios, the TTP may be the VASP's CA or another TTP. These other scenarios would require variants of the protocol presented in this paper.

The algorithms used in the protocol are described in Table 1, while the data elements used are described in Table 2.

The protocol is illustrated in Fig. 2. In the protocol description || represents concatenation. All other symbols are defined in Table 1 and Table 2.

Table 2. Data Elements Used In The Protocol

Data Element	Description
$\boldsymbol{id_U}$	The identity of the user
$\boldsymbol{id_V}$	The identity of the VASP
$\boldsymbol{id_{U\_CA}}$	The identity of the user's CA
$\boldsymbol{K}$	The encryption key generated using the VASP's key agreement key
$\boldsymbol{L}$	The encryption key generated using the TTP's key agreement key
$\boldsymbol{v}$	The private key agreement key of the VASP
$\boldsymbol{g^v}$	The public key agreement key of the VASP
$\boldsymbol{t}$	The private key agreement key of the TTP
$\boldsymbol{g^t}$	The public key agreement key of the TTP
PK_U	The public key of the user used to verify signatures from the user
PK_T	The public key of the TTP used to verify signatures from the TTP
$\boldsymbol{RND_U}$	A random number generated by the user
$\boldsymbol{RND_V}$	A random challenge generated by the VASP
CertU	A valid certificate, issued by a CA, on a public key of the asymmetric signature system of the user
CertV	A valid certificate, issued by a CA, on a public key agreement key g^v of the VASP
$\boldsymbol{cid_U}$	A unique identifier of a certificate on a public key of the asymmetric signature system of the user
$\boldsymbol{cid_V}$	A unique identifier of a certificate on a public key agreement key g^v of the VASP
CertChain(X,Y)	A certificate chain on the public key of *Y* which can be verified by an entity in possession of the public key of the CA of *X* (*X* may take values *U* or *V*, while *Y* may take values *U*, *V*, or *T*, where *U*, *V*, and *T* represent the user, the VASP and the TTP respectively)
$\boldsymbol{\alpha_T}$	The initialisation parameter for the payment scheme
IV	An initialisation vector used to define a family of one-way functions
T	A public system parameter representing the maximum number of ticks which can be committed to using one signature
TS	A time-stamp issued by the TTP
ch_data	The charging data field, typically containing the tariff on which charges should be based

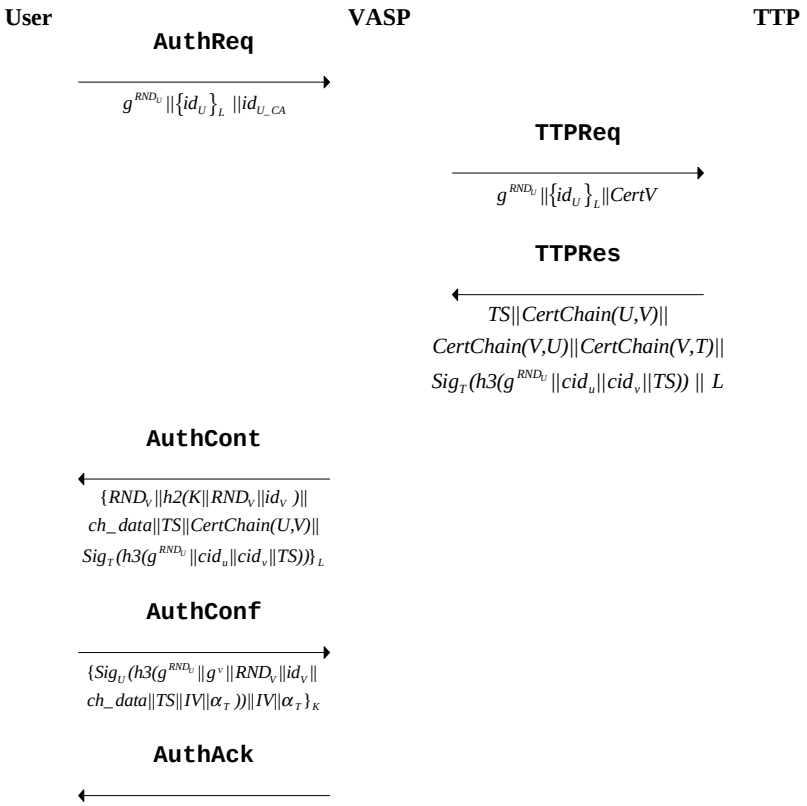

Fig. 2. Authentication And Initialisation Of Charge Ticks Protocol

The authentication and initialisation of charge ticks protocol shown in Fig. 2 works as follows. The user generates a random number RND_U and computes a temporary public key agreement key g^{RNDu}. The user then generates an encryption session key $L = (g^t)^{RNDu}$ using the public key agreement key of the TTP acting as his CA, g^t. He then sends g^{RNDu} to the VASP together with his identity encrypted using L and the identity of the TTP acting as his CA. On receipt of this first message, the VASP contacts the TTP in order to request the appropriate certificates required in the protocol.

The second message is a request to the TTP which will include the encrypted identity of the user and a certificate on the VASP's public key agreement key. The request will also include the parameter g^{RNDu}. This parameter is later cryptographically bound to a time-stamp generated by the TTP and unique identifiers for the certificates used in the protocol. Since a mobile user will not typically have access to a trusted time server, g^{RNDu} gives the user assurance that the time-stamp was created during the current run of the protocol and that the VASP's certificate was not revoked before the start of the current protocol run.

On receiving a request from the VASP, the TTP forms a chain of certificates on the VASP's public key agreement key which the user can verify, and a chain of

certificates on the public keys of the user and the TTP which the VASP can verify. The TTP then sends these back to the VASP in the third message, together with a time-stamp *TS*, the encryption key $L=(g^{RNDu})^t$ and a signature on a hash value of the following parameters: the parameter g^{RNDu}, the unique identifiers of the certificates on the user's and the VASP's public keys respectively, and the time-stamp *TS*.

From the TTP's response message, the VASP uses the certificate chains *CertChain(V,U)* and *CertChain(V,T)* in order to obtain authentic copies of the user's and the TTP's public keys, respectively. The TTP's public key is used to verify the signed data string. *CertChain(U,V)*, *TS* and the signed data string are then forwarded to the user together with the following additional parameters in the fourth message, encrypted using *L*. The first additional parameter is a random challenge RND_V generated by the VASP. The second additional parameter $h2(K||RND_V||id_v)$ is used to confirm possession of the derived session key $K=h1(f((g^{RNDu})^v)||RND_V)=h1(f((g^v)^{RNDu})||RND_V)$. The third additional parameter is the charging data field, *ch_data*.

On receiving the fourth message, the user uses the certificate chain *CertChain(U,V)* to obtain an authentic copy of the VASP's certificate identifier cid_v. This certificate identifier is used together with the certificate identifier of the user's own certificate cid_u, g^{RNDu} and the timestamp *TS*, in order to verify the signed data string. Verifying this data string gives the user assurance that the VASP's certificate, identified by cid_v, was not revoked before the start of the current protocol run. Note that this protocol assumes that the user knows the identifier cid_u of his own certificate.

After checking the acceptability of the time-stamp *TS* and the tariff information contained in the *ch_data* field, the user commits himself to the scheme by including a signed data string in the fifth message of the protocol. As well as *TS* and *ch_data*, the data string which is signed also includes a hash of an initialisation vector *IV* and the *T*-th image of a pre-image chain α_T. The pre-image chain α_T is calculated using a one-way function *F* which is selected from a family of one way functions using the initialisation vector *IV* chosen by the user at the start of each protocol run. The user confirms knowledge of *K* by including a hash of its components g^{RNDu}, g^v and RND_V in the signature. The data in the fifth message of the protocol is encrypted using the session key *K* both as a fundamental part of the authentication protocol and to protect the user identity.

It may be desirable in certain applications to couple the authentication and payment protocol with the information request. In such a scenario, the session key *K* could be used to protect an information field containing the requested Web page address, for example. Such a protected information field could be exchanged during the session as part of the security protocol, triggered by the information request.

7.2 Charge Ticks Protocol

The data elements used in the protocol are described below:

- δ The number of ticks whose payment is requested by the VASP.
- α A pre-image of a hash value corresponding to the number of ticks whose payment is requested by the VASP.

The protocol is illustrated in Fig. 3. In describing the protocol we make the assumption that the tick payment process is ongoing and that in the current data transfer phase the user has been asked by the VASP to pay for δ ticks.

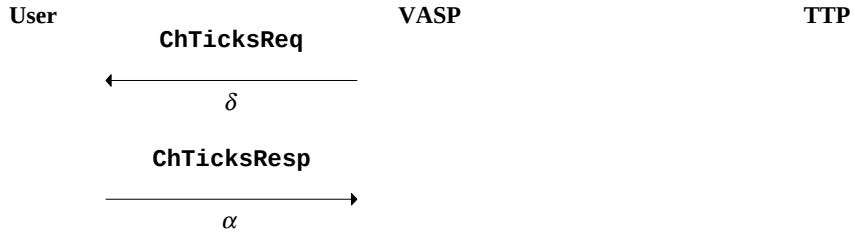

Fig. 3. Charge Ticks Protocol

On receiving a demand for δ ticks from the VASP, the user makes a payment by releasing the appropriate pre-image value α to the VASP, i.e. $\alpha = F^{T-(t+\delta)}(\alpha_0)$, where t is the previous number of ticks released.

7.3 Composition Of Billing Information By VASP

When the call session has ended, or the maximum number of ticks per signature has been reached, the VASP composes and stores the transcript of the charge ticks transaction, representing the bill to be claimed. The information contained in the transcript will include the user's identity id_U, the signature $Sig_U(h3(g^{RNDu}||g^v||RND_v||id_v||ch\_data||TS||IV||\alpha_T))$, the information required to verify the signature g^{RNDu}, g^v, RND_v, id_v, ch_data, TS, IV, α_T, the last received pre-image α, and the number of ticks consumed by the user during the current run of the protocol, *tck_cnt*.

8 Demonstrations And Trials

The focus of the secure billing work in ASPeCT is to investigate the technical feasibility and user acceptability of the proposed charging scheme based on trials and demonstrations. The scheme is thus implemented on PC-based demonstrators and is being trialled over an experimental UMTS platform provided by the ACTS EXODUS project. The ACRYL library from Siemens ZT IK 3 is used for the provision of basic cryptographic functions.

8.1 Selection Of Algorithms And Parameters

Implementation of the scheme involves selecting which particular cryptographic algorithms and parameters should be used. The demonstrators and trials make use of the following algorithms and parameters.

One Way Functions Functions *h1-h3* are implemented using RIPEMD-128 [14, 15] and h4 is implemented using RIPEMD-160 [14, 15], where the output is truncated to 40 and 64 bits for functions *h2* and *h4* respectively.

Exponentiation Exponentiation is conducted in an elliptic curve cryptosystem, whose default parameter values are specified in ISO/IEC 14888-3 [4].

Signature Systems The user uses an elliptic curve based AMV-signature system, as specified in ISO/IEC 14888-3 [4], in order to generate signatures. The TTP may use both an AMV-signature system, as specified in ISO/IEC 14888-3 [4], and an RSA-signature system, as specified in ISO/IEC 9796-2 [3], in order to generate signatures.

Encryption Systems The symmetric encryption system used in the trial is DES in CBC mode [16, 17].

Parameter T The parameter T represents the maximum number of micropayment tokens which can be transferred with respect to a given signed commitment. The default value for T is 2^{10}.

8.2 The ASPeCT Secure Billing Trial

The charging scheme developed in ASPeCT is being trialled over an experimental UMTS platform provided by the ACTS EXODUS project. The secure billing trial involves three ASPeCT entities: a *Mobile User*, a *Value-Added Service Provider* and a *Trusted Third Party*. The EXODUS trial site for the ASPeCT field test provides fixed broadband access to an ATM-based core network. For the purposes of the secure billing demonstrations and trials, the VAS is realised by a World-Wide Web application, where the mobile user is securely billed for information which is downloaded from the VASP's Web server.

The configuration envisaged for the trial is shown in Fig. 4. The ASPeCT software will exist on EXODUS broadband terminal PCs. A software interface will exist which separates ASPeCT and EXODUS functionality. The mobile user will implement some of the ASPeCT security functions on a smart card.

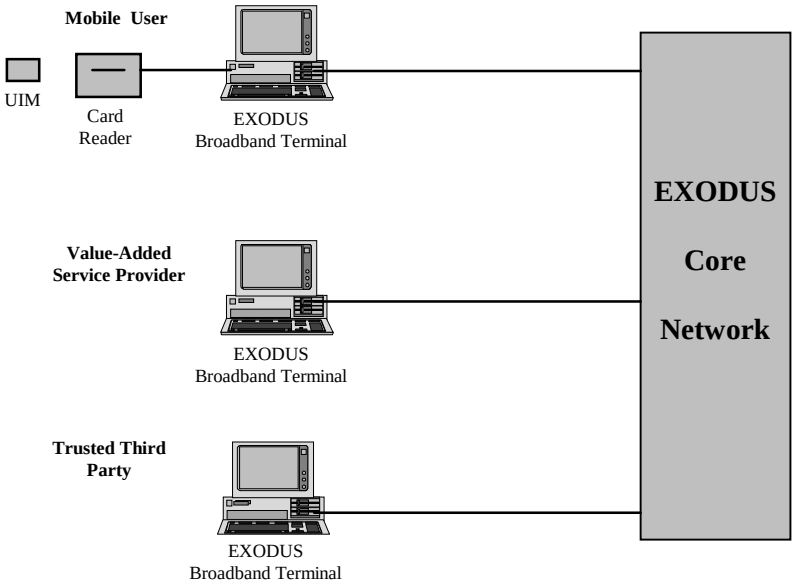

Fig. 4. Trial Configuration

9 Conclusions

We have described a secure billing scheme for value-added information services using micropayments. The scheme makes used of a Trusted Third Party infrastructure in order to certify and manage the public keys required in the charging protocols. Since the scheme is based on an extension of an authentication protocol which has been proposed for UMTS user-to-network authentication, the cost of its introduction is minimised.

The protocol has been optimised for the low bandwidth, storage and processing constraints imposed in a mobile system such as UMTS. As such the design exploits advances in both smart card technology, which make public key cryptography more feasible, and in elliptic curve cryptography, which permits the use of smaller cryptographic parameters. The design also attempts to shift the computational burden away from the mobile user where processing and storage capability are more costly. Furthermore, the design uses non-standard compact public key certificates for efficiency reasons.

More understanding about the applicability and the feasibility of the scheme will be obtained through a trial to be conducted over an experimental UMTS platform provided by the ACTS EXODUS project. This trial will be conducted in Spring 1998.

References

1. ETSI ETS GSM 02.09. European Digital Cellular Telecommunications System (Phase 2); Security Aspects. Version 4.2.4, September 1994.
2. ETSI ETS 300175-7. DECT Common Interface, Part 7: Security Features. October 1992.
3. ISO/IEC 9796-2. Information technology - Security techniques - Digital signature schemes giving message recovery - Part 2: Mechanisms using a hash-function. 1997.
4. ISO/IEC CD 14888-3. Information technology - Security techniques - Digital signature with appendix - Part 3: Certificate-based mechanisms. 1997.
5. T P Pedersen. Electronic payments of small amounts. DAIMI PB-495, Computer Science Department, Aarhus University, August 1995.
6. R L Rivest, A Shamir. PayWord and MicroMint: Two simple micropayment schemes. Cryptobytes Vol 2, No 1, pp7-11, May 1996. Extended version also available from `http://theory.lcs.mit.edu/~rivest`
7. R Anderson, H Manifavas, C Sutherland. A practical electronic cash system. Available from `http://www.cl.cam.ac.uk/users/rja14/`
8. R Hauser, M Steiner, M Waidner. Micro-payments based on iKP. Presented at SECURICOM 96. Available from `http://www.zurich.ibm.com`
9. C S Jutla, M Yung. Paytree: "Amortised-signature" for flexible micropayments. Proceedings of Second USENIX Association Workshop on Electronic Commerce, November 1996, pp213-221.
10. L Lamport. Password authentication with insecure communication. Communications of the ACM, 24:770-772, 1981.
11. R C Merkle. A certified digital signature. Proceedings of CRYPTO '89. Lecture Notes in Comput. Sci., 435:218-238, 1990.
12. ETSI SMG SG DOC 73/95. A public key based protocol for UMTS providing mutual authentication and key agreement. September 1995.
13. L Chen, H J Hitz, G Horn, K Howker, V Kessler, L Knudsen, C J Mitchell, C Radu. The use of trusted third parties and secure billing in UMTS, Proceedings of ACTS Mobile Telecommunications Summit, Granada, November 1996, pp493-499.
14. H Dobbertin, A Bosselaers, B Preneel. RIPEMD-160: a strengthened version of RIPEMD. Fast Software Encryption, Third International Workshop. Lecture Notes in Comput. Sci., 1039:71-82, 1996.
15. ISO/IEC DIS 10118-3. Information technology - Security techniques - Hash-functions - Part 3: Dedicated hash-functions. 1997.
16. FIPS 46. Data Encryption Standard. U.S. Department of Commerce / National Bureau of Standards, Springfield, Virginia, 1977.
17. ISO/IEC 10116. Information technology - Security techniques - Modes of operation for an n-bit block cipher. 1997.